

NASA TM X-66033

A CHARACTER STRING SCANNER

(NASA-TM-X-66033) A CHARACTER STRING
SCANNER R.L. Enison (NASA) Jan. 1971
37 p CSCL 09B

N72-32207

Unclas
G3/08 42071

RICHARD L. ENISON

JANUARY 1971



— GODDARD SPACE FLIGHT CENTER —
GREENBELT, MARYLAND

A CHARACTER STRING SCANNER

by
Richard L. Enison*

January 1971

GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

*Mr. Enison is a coop student from Pratt Institute who is working in the MTAD under the direction of Mrs. I. Cole.

ABSTRACT

A computer program called Character String Scanner (CSS), is presented. It is designed to search a data set for any specified group of characters and then to flag this group. The output of the CSS program is a listing of the data set being searched with the specified group of characters being flagged by asterisks. Therefore, one may readily identify specific keywords, groups of keywords or specified lines of code internal to a computer program, in a program output, or in any other specific data set. Possible applications of this program include the automatic scan of an output data set for pertinent keyword data, the editing of a program to change the appearance of a certain word or group of words, and the conversion of a set of code to a different set of code.

PRECEDING PAGE BLANK NOT FILMED

Preceding page blank

CONTENTS

	<u>Page</u>
1. BACKGROUND	1
2. PURPOSE	1
3. SYMBOLS USED IN THIS REPORT	1
4. METHOD	2
5. INPUT	6
6. OUTPUT	7
7. SAMPLE RUN	9
REFERENCES	10
APPENDIX — THE FORTRAN SUBROUTINE PARM	11

PRECEDING PAGE BLANK NOT FILMED

Preceding page blank

A CHARACTER STRING SCANNER

1. BACKGROUND

It was decided to use a static version of DODS as an interim tool for the R and D user of GTDS. This static version of DODS, however, has to be a version that would allow the R and D user to incorporate his changes on either a temporary or permanent basis. It has also to be a version that the OS/360 does not think is the "real" DODS. Several tools were developed to aid in the development of such a special version of DODS. This paper describes one of these tools.

2. PURPOSE

CSS (Character String Sscanner) scans a character string, e.g. an EBCDIC record of data, in search of a particular group of characters. The character string scanned is printed out, and each occurrence of the group of characters, if any, is flagged below with asterisks. The entire line is flagged with dashes if the group of characters is found. For an example, see the section OUTPUT.

This is done for each record of an input data set, listing the records with or without flags as appropriate, in succession. The user may specify at load time whether the first character of each record is to be ignored as a carriage control character, and how many four-byte words of data are to be scanned.

3. SYMBOLS USED IN THIS REPORT

LUP	— variable containing number of four-byte words to be scanned per record
LCC	— one-byte LOGICAL variable indicating whether first character of each record is to be ignored
PC	— name of COMMON block containing NUM and F
NUM	— variable containing number of characters in PARM field
F	— singly-subscripted LOGICAL*1 type variable containing PARM field
PARM	— subroutine called to access PARM field
DODS	— singly-subscripted INTEGER*2 type variable containing character string 'DODS'

COMPR — singly-subscripted INTEGER*2 type variable containing the group of characters for which to search
 TEXT — singly-subscripted variable into which record is read
 LTEXT — singly-subscripted LOGICAL*1 variable EQUIVALENCED to TEXT
 ITEXT — singly-subscripted INTEGER*2 variable used to break record up into single characters for comparison with COMPR
 ILTEXT — singly-subscripted LOGICAL*1 variable EQUIVALENCED to ITEXT
 ASTER — singly-subscripted variable containing asterisks to be printed, if any
 MARK — singly-subscripted LOGICAL*1 variable EQUIVALENCED to ASTER
 LINE — singly-subscripted variable containing dashes to be printed when appropriate
 IND — one-byte LOGICAL variable indicating whether any flags are to be printed
 N — variable containing the number of characters in COMPR

4. METHOD

The information the user wishes to pass to CSS: 1. whether the first character of each input record is to be ignored, 2. the number of four-byte words to be scanned, and 3. the group of characters for which to search, is passed through a JCL (Job Control Language) feature known as the PARM field. For those readers unfamiliar with this feature, let us review JCL in general.

Every job submitted through 360/OS (Operating System) is defined and invoked by JCL cards. The main types of such cards are JOB, EXEC, and DD cards. The JOB card is the first of every job, and in addition to delimiting jobs, it gives the system such information as the name of the job, of the user, the user's account number, box number and other accounting information, as well as such information as whether JCL cards are to be listed and if so, through what output class.

Now, a job consists of one or more steps. The first card of a step is the EXEC card. This card specifies the program to be executed, together with other information. Data sets used by the program are defined by DD cards immediately following the EXEC card. The first field of the DD card is the ddname, which identifies the information coded on the card for use by system I/O routines. The program does I/O by passing the ddname, along with other information, to the system in an internal Data Control Block (DCB). The data set may exist on disk, drum, data cell, tape, cards or may be sent to a printer or card punch. This

information is coded on the DD card. For example, a card data set is defined by a DD card of the form

```
//ddname DD *
```

.
.
.

data cards

.
.
.

next JCL card

or, if some of the data cards begin with two slashes (//),

```
//ddname DD DATA
```

.
.
.

data cards

.
.
.

```
/*
```

The program invoked by the EXEC statement must itself be in the form of a load module and must be a member of a partitioned data set called a library. This may be the system program library (SYS1.LINKLIB), a private library, or a temporary library which exists only for the duration of the run. An example of the latter is a typical compile-link-go job. The LINK step invokes the linkage editor, which creates a load module, puts it in a temporary library, and then the GO step invokes that. The compilers, assemblers, linkage editors and utility programs provided by IBM all reside in the system library. They are invoked as follows:

```
// EXEC PGM=progname
```

with other parameters optional. Programs in a private library are invoked the same way, except that a DD card defining the library must be provided immediately following the JOB card, and must have the special ddname JOBLIB. Programs in a temporary library are invoked as follows:

```
// EXEC PGM=*.stepname.ddname
```

where stepname is the name of the LINK step in which the load module was created. This is indicated on the EXEC card for that step in the stepname field, which is optional and immediately follows the two slashes:

```
//stepname EXEC ...
```

Frequently used sequences of JCL statements are stored in the system procedure library (SYS1.PROCLIB), which is a partitioned data set whose members are called procedures. Each procedure has one or more EXEC statements, each followed by one or more DD statements. A procedure is invoked as follows:

```
// EXEC procname
```

Parameters on the EXEC cards in the procedure may be overridden by parameters coded on the above EXEC card, identified with the appropriate stepnames as coded in the procedure. Parameters on DD cards in the procedure may be overridden with DD cards immediately following the EXEC card invoking the procedure, with the appropriate stepname coded as part of the ddname on each such card. For example, the procedure FORTRANG invokes the FORTRAN level G compiler, whose name, incidentally, is IEYFORT. The stepname is SOURCE. The following example shows one of each type of override.

```
// EXEC FORTRANG,REGION.SOURCE=300K
```

```
//SOURCE.SYSIN DD *
```

```
.  
.   
.
```

Since FORTRANG has only one step, the use of the stepname SOURCE is unnecessary and may be omitted. An example of a multi-step procedure is LINKGO, which has a LINK and a GO step. The stepnames are LINK and GO.

Among the parameters which may be coded on an EXEC statement is PARM. The parm field is just a character string with at most 100 characters. It is coded as follows:


```
// EXEC ..., PARM='this is the parm field'
```

One may pass information to the compiler, linkage editor, and/or one's own program in a compile-link-go job. An example of all three is:

```
// EXEC FORTRANG, PARM='LIST, DECK'
```

```
// EXEC LINKGO, PARM. LINK='MAP, LIST', PARM. GO='EARTH-MOON'
```

If one passes information in this way to one's own program, then one's program must contain coding to retrieve this information. To do this, one must understand how the system makes this information available to programs.

Upon entry to the program invoked by the EXEC statement, general register 1 contains the address of a pointer to a field of data in core. The field is located on a halfword boundary, and its format is as follows:

	length of field	the field
byte	0	1 2

That is, the first two bytes contain a halfword integer whose value is the length of the PARM field in bytes (characters). This is immediately followed by the information coded in the PARM field itself. CSS, which was written in FORTRAN, calls a FORTRAN SUBROUTINE called PARM which accesses this information and returns it in the COMMON block PC. This SUBROUTINE is described in the appendix.

The PARM field retrieved as above is interpreted by CSS as having three fields or less. The formats of these fields is described in the section INPUT. The three fields are:

1. A logical value, true or false, indicating whether the first character of each record is to be ignored.
2. A number, from 1 to 33, indicating the number of words of data to be scanned per record.
3. A character string, indicating the group of characters for which to search.

If one or more fields are omitted, default values are assumed. The default for field one is true, for field two 30, and for field three 'DODS'.

The values of these three fields, whether default or explicitly specified, are placed in the variables LCC, LUP and COMPR, respectively. The variable N is set equal to the number of characters in COMPR. LUP and N, or functions thereof, are used as upper limits in DO loops related to the scan. The logical switch LCC is used to decide which FORMAT statement should be used to read the record. In the READ statement, an iterated I/O list is used, with LUP as the upper limit, to read the desired number of words into the elements of TEXT. These elements are then printed. Then in a DO loop, MARK is initialized as all blanks, and TEXT is broken up into characters by moving the elements of LTEXT into every other element of ITEXT. The odd elements have already been initialized as blanks in a specification statement. Thus, upon exit from the loop, ITEXT will be an array of halfwords each of the form

	blank	character
byte	0	1

That is, the first byte is blank and the second, a character from the record. This can be compared with the elements of COMPR, which have similar form. This is done because, with some compilers, one cannot compare logical variables for equality.

Next, IND is set to FALSE, so that if its value is not changed during the scan by finding a match, it will be FALSE after the scan and so ASTER and LINE will not be printed. Finally, we have the scan itself, which consists of a DO loop with two inner loops. The outer loop is indexed by the number of the character in ITEXT which is the first of those we are comparing with COMPR. The first inner loop carries out the comparison, character by character, up to N iterations. If a mismatch is found, control is passed to the end of the outer loop. If not, the second inner loop is executed, which sets the appropriate elements of MARK to asterisks. Then IND is set to TRUE, and we drop to the end of the outer loop. Upon exit from the outer loop, we check IND. If it is TRUE, we print ASTER and LINE and branch back to statement 50 where the test on LCC is made and another record is read in. Otherwise, we go directly to statement 50. When the last record has been processed, we take the END exit of the READ statement and STOP.

5. INPUT

CSS accepts two inputs: the PARM field, and a sequential data set of records to be scanned. The latter is read on logical FORTRAN unit 8, so it is defined in

the GO step by a DD card with the ddname FT08F001. It need have no special format, except that no record may be shorter than LUP words. It is suggested that a fixed record format be used, to prevent some records, such as pagination records, from being too short.

The format of the PARM field is:

	logical switch	ignored	number of words	ignored	group of characters
byte	1	2	3	4 5	6

where logical switch is either T or F, for true or false respectively, indicating whether the first character is to be ignored; number of words is a two-digit decimal numeral indicating the number of words of data to be scanned per record — if the number is less than 10, the first digit must be coded as 0; and group of characters is the character string for which to search. All fields must be entered in the positions shown. The ignored characters may be used optionally as punctuation or separation to make the field readable. I prefer commas.

Fields may be omitted by shortening or omitting the PARM field. If number of words is coded, so must logical switch, and if group of characters is coded, these two must be also. If no PARM field is coded, all there quantities will assume their default values. If a PARM field is coded and one or two fields are omitted, the cutoff must be immediately before one of the ignored characters. To include the ignored character as the last character in the PARM field will produce unpredictable results. For an example, the following PARM field explicitly defines the default values:

```
// EXEC LINKGO, PARM. GO='T, 30, DODS'
```

6. OUTPUT

The only output of CSS is a listing of the input, or of the input records truncated by specification of number of words, together with flags when appropriate, as explained in the section PURPOSE. For example, suppose we are searching for the characters 'MOON' in a card data set (deck). Then the JCL would be:

JOB CARD

.
.
.

```
// EXEC LINKGO, PARM. GO='F,20,MOON'
```

```
.
```

```
.
```

```
.
```

```
//GO.FT08F001 DD *
```

```
.
```

```
.
```

```
.
```

DC

EPHEM

EARTH-MOON DISTANCE

SATELLITE VELOCITY

POTENTIAL OF MOON AT PERIGEE, 500 KM FROM MOON

```
.
```

```
.
```

```
.
```

/*

and the output would be:

```
.
```

```
.
```

```
.
```

DC

EPHEM

EARTH-MOON DISTANCE

* * * *

SATELLITE VELOCITY

POTENTIAL OF MOON AT PERIGEE, 500 KM FROM MOON

* * * *

* * * *

7. SAMPLE RUN

In the run shown at the end of this document, we wish to edit a MAPDISK listing of the disk pack DODS01, flagging all occurrences of 'DODS'. No PARM field is coded, since we want the default values. The first step invokes MAPDISK, which resides in the system library. The output is placed in the data set defined by the DD card with ddname SYSPRINT. This is ordinarily sent to a printer, but we want it to be processed by CSS in a later step, so we code a DD card defining a temporary sequential data set on disk. The remaining steps are just a compile-link-go of CSS. In the GO step, we code the FT08F001 DD card to retrieve the data set created in the MAP step.

This run is typical of a larger class of uses of CSS, but many others are possible, with correspondingly varied JCL deck setups. To learn how to set up the JCL for a particular application of CSS, the reader is referred to reference #1. The other references may provide the reader with a background for understanding how CSS works, depending upon his familiarity with IBM System/360 and its Operating System.

REFERENCES

1. IBM System/360 Operating System, Job Control Language. IBM Corporation, Form C28-6593
2. IBM System/360 Operating System, Supervisor and Data Management Services. IBM Corporation, Form C28-6646
3. IBM System/360, Principles of Operation. IBM Corporation, Form A22-6821
4. IBM System/360 Operating System, Assembler Language. IBM Corporation, Form C28-6514
5. IBM System/360 Operating System, Fortran G and H Programmers' Guide. IBM Corporation, Form C28-6817

APPENDIX

THE FORTRAN SUBROUTINE PARM

PURPOSE

The SUBROUTINE PARM was written to retrieve the PARM field for CSS. There is no reason, however, why it could not perform this service for any other routine. Some caution should be exercised by the would-be user, as there are several restrictions on its use. Here is a list of such restrictions:

1. PARM must be called directly by the MAIN routine.
2. The CALL statement must be the first executable statement in the routine.
3. The CALL statement must have no argument list, i. e., it must be of the form

CALL PARM

The number of characters in the PARM field is returned in an INTEGER variable called NUM, and the characters of the PARM field themselves are returned in the one-byte elements of the LOGICAL array F, both in the COMMON block PC. Thus, in order to have access to this information, the user must code these specification statements:

```
LOGICAL*1 F(100)
COMMON/PC/NUM, F
```

Of course, the names NUM and F are arbitrary and may be changed by the user. If there is no PARM field, NUM will have the value 0. In any case, the first NUM elements of F will contain the characters of the PARM field, one per element, and the remaining elements of F will have unpredictable contents.

LIST OF SYMBOLS USED IN PARM

PC — COMMON block containing NUM and F
NUM — variable in which length of PARM field is returned

F — singly-subscripted LOGICAL*1 variable in which PARM field is returned

ARRAY — singly-subscripted LOGICAL*1 variable containing system PARM field, including length field

INUM — INTEGER*2 variable into which length is moved

LN — singly-subscripted LOGICAL*1 variable EQUIVALENCED to INUM

METHOD

The SUBROUTINE statement has an argument list with the single argument ARRAY. Now the way argument lists are passed in FORTRAN under IBM System/360 OS is as follows: the address of the argument list is placed in general register 1 by the calling routine. The argument list itself is located on a word boundary and consists of a sequence of one or more single-word address constants. In FORTRAN these would be the addresses of variables. Thus the SUBROUTINE PARM will expect, upon entry, to find in register 1 the address of a one-argument argument list pointing to an array. References in the SUBROUTINE to elements of ARRAY will retrieve the data in the field pointed to by the address constant. Now, since the CALL statement was the first statement executed in the program and since it had no argument list, register 1 will contain whatever it had on entry to the program, namely, the address of a pointer to the system PARM field, which includes a length field. All that needs to be done, is to move the information, in appropriate format, into the COMMON variables NUM and F.

The first two bytes of the system PARM field are the length, in bytes, of the PARM field itself. We move these bytes, the first two elements of ARRAY, into the two elements of LN. This is EQUIVALENCED to the halfword integer variable INUM, which therefore now has as its value the length of the PARM field. We next set the fullword integer variable NUM equal to INUM, and half the job is done. If NUM is 0, it means there is no PARM field and we RETURN. Otherwise, we execute a DO loop indexed by NUM to move the elements of ARRAY from the third on into the elements of F. NUM and F now contain all the necessary information, so we RETURN.

SOURCE LISTING

A source listing of PARM, as well as of CSS, can be found in the SAMPLE RUN below.

IEF2531 GFRLECSS SYSOLTB.

//GFRLECSS JOB (GECC31385F,T,300203,005005),CCC,MSGLEVEL=1

//MAP EXEC PGM=MAPDISK

//SYSPRINT DD DSN=DLIST,DISP=(NEW,PASS),DCB=(RECFM=FB,LRECL=133,

// BLKSIZE=7132),SPACE=(TRK,(5,1)),UNIT=2314

//SYSUT1 DD UNIT=2314,VOL=SER=G1SCR2,DISP=SHR

//DDN DD UNIT=2314,VOL=SER=DDDS01,DISP=SHR

//SYSABEND DD SYSOUT=A,SPACE=(CYL,(1,1))

IEF2351 ALLOC. FOR GFRLECSS MAP

IEF2371 541 ALLOCATED TO SYSPRINT

IEF2371 231 ALLOCATED TO SYSUT1

IEF2371 544 ALLOCATED TO DDN

IEF2371 541 ALLOCATED TO SYSABEND

IEF2851 SYS7C274.T154519.RV000.GFRLECSS.LIST PASSED

IEF2851 VOL SER NOS= G1SCR3.

IEF2851 SYS7C274.T154519.RV000.GFRLECSS.H0000158 KEPT

IEF2851 VOL SER NOS= G1SCR2.

IEF2851 WORK.SKCLIB KEPT

IEF2851 VOL SER NOS= DDDS01.

IEF2851 SYS7C274.T154519.SV000.GFRLECSS.H0000160 DELETED

IEF2851 VOL SER NOS= G1SCR3.

-----JOB NBR= 090 STEP NBR= 01 GFRLECSS MAP PGM=MAPDISK CARDS=00000 INITIATION TIME=21.09.13.25 DATE=10-01-70

-----CPU=000.1 I/O=000.1 CORE=000.1 CHARGE=000.09 STEP=01 MAP TERMINATION TIME=21.10.37.71 DATE=10-01-70

-----I/O TIME BY DEVICE. DISK=*****.11,DRUM=*****.01,TAPE=*****.00,CELL=*****.00,UTH=*****.00

-----STEP REGION SIZE=000K MAXIMUM REGION SIZE USED=0042K PERCENT OF REGION USED=52

//PEEK EXEC FORTRANS,PARM=DECK

XXDEFAULT PROC VOLK=20

00000100

XXSOURCE EXEC PGM=IEYFUT,REGION=200K

00000200

XXSYSLIN DD DSN=DDDSJMOD,UNIT=DISK,SPACE=(3200,(6,RELK,4)),ROUND),

00000300

IEF6531 SUBSTITUTION JCL - DSN=DDDSJMOD,UNIT=DISK,SPACE=(3200,(20,4)),ROUND),

XX DISP=(MOD,PASS),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)

00000400

XXSYSPRINT DD SYSOUT=A,DCB=(RECFM=FB,LRECL=120,BLKSIZE=7200),

00000500

XX SPACE=(CYL,(2,1))

00000600

//SOURCE.SYSPUNCH DD DSN=DECK,SYSOUT=B

IEF6531 SUBSTITUTION JCL - DSN=DECK,SYSOUT=B

X/SYSPUNCH DD DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=7200),

00000700

XX SPACE=(TRK,(10,5))

00000800

IEF6551 DSN=INVALID WHEN SYSOUT SPECIFIED

//SOURCE.SYSABEND DD SYSOUT=A,SPACE=(CYL,(1,1))

//SOURCE.SYSIN DD *

IEF2361 ALLOC. FOR GFRLECSS SOURCE PEEK

IEF2371 540 ALLOCATED TO SYSLIN

IEF2371 541 ALLOCATED TO SYSPRINT

IEF2371 541 ALLOCATED TO SYSPUNCH

IEF2371 541 ALLOCATED TO SYSABEND

IEF2371 541 ALLOCATED TO SYSIN

```

0001      I T E G E R * 2 I T E X T ( 1 3 2 ) / 1 3 2 * ' ' / , D C C S ( 4 ) ' ' C ' , ' C ' , ' S ' / ,
      A S T 2 ' ' ' / , I T E L / ' ' ' / , I T E M P / 0 ' , J T E M P / ' ' / , C C M P R ( ' > 5 ) , 8 8 ' / ' ' /
0002      L O G I C A L * 1 L T E X T ( 1 3 2 ) , I L T E X T ( 2 6 4 ) , M A R K ( 1 3 2 ) , E 1 , A S T 1 , I N D , L C C / , T R U E . /
0003      L O G I C A L * 1 F ( 1 0 0 ) , ' ' ' ' L T E M P ( 2 ) , L J T E M P ( 2 )
0004      D I M E N S I O N T E X T ( 3 3 ) , A S T E R ( 3 3 ) , L I N E ( 3 3 )
0005      D A T A L I N E ' / 3 3 * ' ' ' ' ' / , A S T E R / 3 3 * ' ' ' ' ' / , L U P / 3 0 / ' ' ' ' ' /
0006      E Q U I V A L E N C E ( T E X T ( 1 ) , L T E X T ( 1 ) ) , ( I T E X T ( 1 ) , I L T E X T ( 1 ) ) , ( B 1 , D O O S ( 1 ) )
      ' ' , ( A S T 1 , A S T 2 ) , ( A S T E R ( 1 ) , M A R K ( 1 ) ) , ' ' ' ' ( I T E M P , L T E M P ( 1 ) )
      E Q U I V A L E N C E ( J T E M P , L J T E M P ( 1 ) )
0007      C O M M O N / P C / N L W , F
0008      C A L L P A R M
0009      I F ( N U M , E Q , 0 ) G C T C 4 0
0010      L J T E M P ( 1 ) = F ( 1 )
0011      I F ( J T E M P , N E , I T E L ) L C C = F A L S E .
0012      L T E M P ( 2 ) = F ( 3 )
0013      I T E M P = I T E M P - 2 4 0
0014      L L P = 1 0 * I T E M P
0015      L T E M P ( 2 ) = F ( 4 )
0016      L L P = L L P + I T E M P - 2 4 0
0017      4 0 L L P = 4 * L L P
0018      N = 4
0019      D O 7 0 K = 1 , 4
0020      7 0 C C M P R ( K ) = D O O S ( K )
0021      I F ( N U M , L E , 4 ) G C T C 4 5
0022      N = N U M - 5
0023      J T E M P = 0 2
0024      D O 8 0 K = 1 , N
0025      L J T E M P ( 2 ) = F ( K + 5 )
0026      8 0 C C M P R ( K ) = J T E M P
0027      4 5 N M 1 = N - 1
0028      L L P 4 M 3 = L U P 4 - N M 1
0029      5 0 I F ( L C C ) G C T C 5 1
0030      R E A D ( 8 , 1 0 0 , E N D = 3 0 ) ( T E X T ( K ) , K = 1 , L U P )
0031      1 0 0 F O R M A T ( 3 3 A 4 )
0032      G O T O 5 2
0033      5 1 R E A D ( 8 , 1 0 1 , E N D = 3 0 ) ( T E X T ( K ) , K = 1 , L U P )
0034      5 2 W R I T E ( 6 , 1 0 1 ) ( T E X T ( K ) , K = 1 , L U P )
0035      1 0 1 F O R M A T ( 1 X , 3 3 A 4 )
0036      D O 1 0 K = 1 , L L P 4
0037      M A R K ( K ) = B 1
0038      1 0 I L T E X T ( 2 * K ) = L T E X T ( K )
0039      I C = 0
0040      I N D = F A L S E .
0041      D O 2 0 K = 1 , L L P 4 M 3
0042      I F ( I C ) 1 1 , 1 1 , 1 9
0043      1 1 D O 9 0 K K = 1 , N
0044      I F ( I T E X T ( K + K K - 1 ) , N E , C C M P R ( K K ) ) G C T C 2 0
0045      9 0 C O N T I N U E
0046      I C = N
0047      D O 2 0 0 K K = 1 , N
0048      2 0 0 M A R K ( K + K K - 1 ) = A S T 1
0049      I N D = T R U E .
0050      1 9 I C = I C - 1
0051      2 0 C O N T I N U E
0052      I F ( I N D ) W R I T E ( 5 , 1 0 2 ) A S T E R , L I N E
0053      1 0 2 F O R M A T ( 1 X , 3 3 A 4 / 1 X , 3 3 A 4 / )

```

FURTRAN IV G LEVEL 12

MAIN

DATE = 70274

21/10/13

PAGE 0002

0055 GO TO 50
0056 SC STOP
0057 END

COMMON BLOCK / PC / MAP SIZE 63

SYMBOL NUM	LOCATION 0	SYMBOL F	LOCATION 4	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
---------------	---------------	-------------	---------------	--------	----------	--------	----------	--------	----------

SUBPROGRAMS CALLED

SYMBOL PARM	LOCATION 118	SYMBOL IBCCM=	LOCATION 11C	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
----------------	-----------------	------------------	-----------------	--------	----------	--------	----------	--------	----------

EQUIVALENCE DATA MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
TEXT	120	TEXT	120	ITEXT	1A4	ITEXT	1A4	BI	2AC
DODS	2AC	AST1	2B4	AST2	2E4	ASTER	20E	MARK	208
ITEMP	33C	ITEMP	33C	ITEMP	33C	ITEMP	33C		

SCALAR MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
LUP	340	LUP4	344	N	345	K	34C	NM1	350
LUP4M3	354	IC	356	KK	35C	ITEE	36C	B2	362
LCC	364	INC	365						

ARRAY MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
COMPR	366	LINE	424						

FORMAT STATEMENT MAP

SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION	SYMBOL	LOCATION
100	4A8	101	4AE	102	4EB				

OPTIONS IN EFFECT NOID, EDCDIC, SOURCE, NCLIST, DECK, LEAD, MAP

OPTIONS IN EFFECT NAME = MAIN, LINECNT = 53

STATISTICS SOURCE STATEMENTS = 07, PROGRAM SIZE = 2230

STATISTICS NO DIAGNOSTICS GENERATED

```
0001      SUBROUTINE PARM(ARRAY)
0002      LOGICAL*1 F(100),ARRAY(102),LN(2)
0003      INTEGER*2 INCM
0004      EQUIVALENCE (INCM,LN(1))
0005      COMMON /PC/NUM,F
0006      LN(1)=ARRAY(1)
0007      LN(2)=ARRAY(2)
0008      NCM=INCM
0009      IF (NUM.EQ.0) RETURN
0010      DO 10 K=1,NCM
0011      10 F(K)=ARRAY(K+2)
0012      RETURN
0013      END
```

FORTRAN IV G LEVEL 18

PARM

DATE = 70274

21/10/13

PAGE 0002

COMMON BLOCK / PC		/ MAP SIZE 88	
SYMBOL	LOCATION	SYMBOL	LOCATION
NUM	0		

EQUIVALENCE DATA MAP			
SYMBOL	LOCATION	SYMBOL	LOCATION
INUM	A0	LN	A0

SCALAR MAP			
SYMBOL	LOCATION	SYMBOL	LOCATION
K	AA		

ARRAY MAP			
SYMBOL	LOCATION	SYMBOL	LOCATION
ARRAY	AB		

OPTIONS IN EFFECT NOID,EDD,ICYSOURCE,INCLIST,DECK,EGAG,MAP

OPTIONS IN EFFECT NAME = PARM , LINECNT = 58

STATISTICS SOURCE STATEMENTS = 13, PROGRAM SIZE = 418

STATISTICS NO DIAGNOSTICS GENERATED

STATISTICS NO DIAGNOSTICS THIS STEPA

```

IEF2851 SYS70274.T154519.RV000.GFRLECSS.CBJMUD PASSED
IEF2851 VOL SER NOS=GISCR4.
IEF2851 SYS70274.T154519.SVC00.GFRLECSS.R0000161 SYSCUT
IEF2851 VOL SER NOS=GISCR5.
IEF2851 SYS70274.T154519.SV000.GFRLECSS.R0000162 SYSCUT
IEF2851 VOL SER NOS=GISCR5.
IEF2851 SYS70274.T154519.SV000.GFRLECSS.R0000163 DELETED
IEF2851 VOL SER NOS=GISCR9.
IEF2851 SYS70274.T154519.RV000.GFRLECSS.S0000164 SYSIN
IEF2851 VOL SER NOS=GISCR5.
IEF2851 SYS70274.T154519.RV000.GFRLECSS.S0000164 DELETED
IEF2851 VOL SER NOS=GISCR5.
-----JOB NBR= 690 STEP NBR= 02 GFRLECSS SOURCE PGM=IEYFRT CARCS=00072 INITIATION TIME=21.10.07.89 DATE=10-01-70
-----CPU=000.1 I/O=000.1 CORE=000.1 CHARGE=000.02 STEP=02 SOURCE TERMINATION TIME=21.10.24.01 DATE=10-01-70
----- I/O TIME BY DEVICE. DISK=*****.30,DRUM=*****.09,TAPE=*****.00,CELL=*****.00,UTR=*****.00
-----STEP REGION SIZE=0200K MAXIMUM REGION SIZE USED=0130K PERCENT OF REGION USED=65
//LG EXEC LINKGO
XXDEFAULT PROC NBLK=5C 00000100
XXLINK EXEC PGM=IE*LINK,PARM=(MAP,LIST),CCND=(5,LT),REGION=300K 00000200
XXLOADLIB DD DSN=SYS2.LOADLIB,DISP=SHR 00000300
XXNEWLIB DD DUMMY 00000400
XXSYSLIB DD DSN=SYS2.DUMMY,DISP=SHR 00000500
XX DD DSN=SYS2.DUMMY,DISP=SHR 00000600
XX DD DSN=SYS1.PORTLIB,DISP=SHR 00000700
XX DD DSN=SYS2.GSFCLIB,DISP=SHR 00000800
XX DD DSN=SYS1.PLIB,DISP=SHR 00000900
XX DD DSN=SYS1.TELCLIB,DISP=SHR 00001000
XX DD DSN=SYS2.LOADLIB,DISP=SHR 00001100
XX DD DSN=SYS1.SSPAK,DISP=SHR 00001200
XXSYSLMOD DD DSN=SYS2.SYSLMOD(GSFC),DISP=(NEW,PASS),UNIT=DISK, 00001300
XX SPACE=(3072,(5,NBLK,40,1)),DCB=BLKSIZE=3072 00001400
IEF2851 SUBSTITUTION JCL -- SPACE=(3072,(50,40,1)),DCB=BLKSIZE=3072
XXSYSPRINT DD SYSOUT=A,DCB=(RECFM=FB,M,LRECL=121,BLKSIZE=1210), 00001500
XX SPACE=(1KK,(3,3)) 00001600
XXSYSLIB DD UNIT=DISK,SPACE=(1024,(100,20)),DCB=BLKSIZE=1024 00001700
XXTAPELIB DD DUMMY,VOL=SER=TAPEIN,UNIT=(9TRACK,DEFER),LABEL=(,ELP), 00001800
XX DISP=(OLD,KEEP),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200) 00001900
XXSYSLIN DD DSN=SYS2.CBJMUD,DISP=(OLD,DELETE),DCB=RECFM=FB 00002000
XX DD DNAME=OBJECT 00002100
//LINK.SYSABEND DD SYSOUT=A,SPACE=(CYL,(1,1))
IEF2361 ALLOC. FOR GFRLECSS LINK LG
IEF2371-337 ALLOCATED TO LOADLIB
IEF2371 237 ALLOCATED TO SYSLIB
IEF2371-237 ALLOCATED TO
IEF2371 100 ALLOCATED TO
IEF2371-237 ALLOCATED TO
IEF2371 237 ALLOCATED TO
IEF2371-237 ALLOCATED TO
IEF2371 337 ALLOCATED TO
IEF2371-337 ALLOCATED TO
IEF2371 332 ALLOCATED TO SYSLMOD
IEF2371-332 ALLOCATED TO SYSPRINT
IEF2371 332 ALLOCATED TO SYSLIB
IEF2371-332 ALLOCATED TO SYSLIN
IEF2371 332 ALLOCATED TO SYSABEND

```

F128-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED MAP,LIST
 VARIABLE OPTIONS USED SIZE=(256000,6144)

DEFAULT OPTION(S) USED

MODULE MAP

CONTROL SECTION

ENTRY

NAME	ORIGIN	LENGTH	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION	NAME	LOCATION
MAIN	00	28C								
PARM	9C0	1A2								
IHCECOM1*	A68	F31								
IHCECOM1E*	19A0	581	IBCCM=	A00	FDIUCS=	B24	INTSWCH	1986		
IFCFVTH*	1F28	1155	SEGDA5D	1C3C						
			ADCCN=	1F25	FCVAOUTP	1FD2	FCVLJUTP	2082	FCVZOUTP	2182
			FCVOUTP	2553	FCVEOUTP	2A5A	FCVCOUTP	2C74	INT6SWCH	2F58
IHCFNTH*	30C0	512	ANITH=	30C0	ADJSWCH	3A2C				
IHCFIOS*	35D8	129C								
IFCUOPT *	4878	350	FIGCS=	3503	FIGCSBP	360E				
IHGERRM *	48C6	58C								
IFCUATOL*	518E	638	ENRMCK	4EC3	IFCERRE	48E0				
IHCETRCH*	57C0	28E								
PC	5A5C	68	INSTRCH	57C0	ENHTRA	57C8				

ENTRY ADDRESS 0C
 TOTAL LENGTH 5AB6

****GSFC DOES NOT EXIST BUT HAS BEEN ADDED TO DATA SET


```

IEF2851  SYS2.LOADLIB                                KEPT
IEF2851  VOL SER NOS=GISYS2.
IEF2851  SYS2.DUMMY                                KEPT
IEF2851  VOL SER NOS=GISYS1.
IEF2851  SYS2.DUMMY                                KEPT
IEF2851  VOL SER NOS=GISYS1.
IEF2851  SYS1.FORTLIB                                KEPT
IEF2851  VOL SER NOS=GI0RM1.
IEF2851  SYS2.GSFCLIB                                KEPT
IEF2851  VOL SER NOS=GISYS1.
IEF2851  SYS1.PL1LIB                                KEPT
IEF2851  VOL SER NOS=GISYS1.
IEF2351  SYS1.TELCLIB                                KEPT
IEF2851  VOL SER NOS=GISYS1.
IEF2851  SYS2.LOADLIB                                KEPT
IEF2851  VOL SER NOS=GISYS2.
IEF2851  SYS1.SSPAK                                KEPT
IEF2851  VOL SER NOS=GISYS2.
IEF2851  SYS70274.T154519.RV000.GFRLECSS.LODNCD    PASSED
IEF2851  VOL SER NOS=GISCR9.
IEF2851  SYS70274.T154519.SV000.GFRLECSS.R0000165  SYSCUT
IEF2851  VOL SER NOS=GISCR9.
IEF2851  SYS70274.T154519.RV000.GFRLECSS.R0000166  DELETED
IEF2851  VOL SER NOS=GISCR9.
IEF2851  SYS70274.T154519.RV000.GFRLECSS.08JMOD    DELETED
IEF2851  VOL SER NOS=GISCR4.
IEF2851  SYS70274.T154519.SV000.GFRLECSS.H0000167  DELETED
IEF2851  VOL SER NOS=GISCR9.
-----JOB NBR= 690 STEP NBR= 03 GFRLECSS LINK PGM=IEWL CARDS=00000 INITIATION TIME=21.10.24.76 DATE=10-01-70
-----CPU=C0001 I/O=C0001 CORE=0001 CHARGE=000.00 STEP=03 LINK TERMINATION TIME=21.10.55.44 DATE=10-01-70
-----I/O TIME BY DEVICE. DISK=*****1.06,DRUM=*****.60,TAPE=*****.00,CELL=*****.00,OTIM=*****.00
-----STEP REGION SIZE=0300K MAXIMUM REGION SIZE USED=0254K PERCENT OF REGION USED=84
XXGO EXEC PGM=LINK,SYSLMOD,CCND=(S,LT),REGION=116K 00002200
XXFT03F001 DD DDNAME=DATAS 00002300
XXFT06F001 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7255), 00002400
XX SPACE=(CYL,(1,1)) 00002500
XXFT07F001 DD DUMMY,DCB=(RECFM=FB,LRECL=30,BLKSIZE=7250), 00002500
XX SPACE=(TRK,(1,20)) 00002600
*** INSERT //GD.FT(7F001 DD DSN=88DECK,SYSCUT=8 FOR PUNCHED OUTPUT 00002700
XXSYSPRINT DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=625), 00002800
XX SPACE=(TRK,(1,20)) 00002900
//GO.FT08F001 DD DSN=6LIST,DISP=(OLD,DELETE)
//GO.SYSABEND DD SYSOUT=A,SPACE=(CYL,(1,1))
//
IEF2361 ALLOC. FOR GFRLECSS GD LG
IEF2371 332 ALLOCATED TO PGM=GD
IEF2371 333 ALLOCATED TO FT06F001
IEF2371 333 ALLOCATED TO SYSPRINT
IEF2371 541 ALLOCATED TO FT08F001
IEF2371 333 ALLOCATED TO SYSABEND

```

GSFC SYSTEMS SUPPORT UTILITY BASIC ALLOCATION MAP UPDATED 5/12/68

PAGE

CONTENTS ON VOLUME=SER=000501 UNIT=54

VTDC	DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	FILE EXTENTS	FILE SERIAL	VOL. SEQ.	SECURITY	TOTAL ALLUC	TRACKS USED
	EXT--FIRST--LAST-LENGTH								10	3
	01-00001-00010-00010									

FREE SPACE

EXT--FIRST---LAST-LENGTH

C1 00390 C043E CC057
 C2 01026 C1054 00057
 C3 01173 C1175 CC001
 C4 01616 C1514 00005
 C5 01636 C16E5 CC050
 C6 02128 C2162 00055
 C7 02420 C2435 CC020
 C8 02476 C2479 CC002
 C9 02816 C2817 00002
 C10 03100 C3158 CC039
 C11 03955 C3557 CC003
 C12 03999 C3555 00001

AAOBS

69255 79255 PART 01 000501 C1 NO 30 29

DSORG=PD RECFM=L LRECL=0

BLKSIZE=7694 2ND ALLOCATION=5

EXT--FIRST---LAST-LENGTH

C1 00440 00465 00030

AOSFL

69255 00000 SEQ. 01 000501 C1 NO 17 17

DSORG=PS RECFM=F LRECL=0

BLKSIZE=123 2ND ALLOCATION=0

EXT--FIRST---LAST-LENGTH

C1 02666 02676 00017

ARC.WORKLIB

70176 00000 PART 01 000501 C1 NO 20 8

DSORG=PD RECFM=FB LRECL=30

BLKSIZE=3200 2ND ALLOCATION=0

EXT--FIRST---LAST-LENGTH

C1 01680 01685 CC030

ATCT51

70072 00000 SEQ. 01 000501 C1 NO 1 1

DSORG=PS RECFM=V LRECL=612

BLKSIZE=616 2ND ALLOCATION=0

EXT--FIRST---LAST-LENGTH

C1 03155 03155 00001

AUTOBS

70255 00000 SEQ. 03 000501 C1 NO 251 251

DSORG=PS RECFM=F LRECL=0

BLKSIZE=3520 2ND ALLOCATION=5

EXT--FIRST---LAST-LENGTH

C1 00634 C0135 CC109

C2 02153 C2322 CC130

C3 03740 C3754 CC015

C64ELS

70255 00000 SEQ. 01 000501 C1 NO 221 221

DSORG=PS RECFM=F LRECL=0

BLKSIZE=3520 2ND ALLOCATION=5

EXT--FIRST---LAST-LENGTH

C1 01560 C2035 CC030

C2 02677 C2755 CC079

C3 03755 C3816 CC032

GSFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SEN=000S01 UNIT=544

DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	EXTENTS	FILE SERIAL	VOL. SEQ.	SECURITY	TOTAL ALLOC	TRACKS USED
C64MNV	70265	00000	SEQ.	01	000S01 ****	01	NU	10	10
DSORG=PS RECFM=F LRECL=0 BLKSIZE=3520 2ND ALLOCATION=2 EXT--FIRST--LAST-LENGTH 01 02953 02563 00010									
DCPSCD	70255	00000	SEQ.	01	000S01 ****	01	NU	11	0
DSORG=PS RECFM=VS LRECL=604 BLKSIZE=608 2ND ALLOCATION=0 EXT--FIRST--LAST-LENGTH 01 00023 00033 00011									
D003.AIL18E ****	69250	79250	PART	02	000S01 ****	01	NU	4	4
DSORG=PS RECFM=L LRECL=3072 BLKSIZE=7254 2ND ALLOCATION=1 EXT--FIRST--LAST-LENGTH 01 01176 01178 00003 02 01130 01136 00001									
D003.DML18 ****	69250	79250	PART	01	000S01 ****	01	NU	2	2
DSORG=PS RECFM=L LRECL=3072 BLKSIZE=3072 2ND ALLOCATION=1 EXT--FIRST--LAST-LENGTH 01 00011 00012 00002									
D003.DUMMY ****	69250	00000	PART	01	000S01 ****	01	NU	1	1
DSORG=PS RECFM=L LRECL=0 BLKSIZE=1024 2ND ALLOCATION=0 EXT--FIRST--LAST-LENGTH 01 01130 01132 00001									
D003.DUMPS ****	70152	99330	PART	00	000S01 ****	01	NU	0	
DSORG= PS RECFM=L LRECL=0 BLKSIZE=7254 2ND ALLOCATION=0 EXT--FIRST--LAST-LENGTH									
D003.GUMOU ****	70170	74000	PART	02	000S01 ****	01	NU	260	246
DSORG=PS RECFM=L LRECL=0 BLKSIZE=7254 2ND ALLOCATION=20 EXT--FIRST--LAST-LENGTH 01 00180 00375 00200 02 03100 03239 00050									
D003.MODL13	70170	00000	PART	02	03SCR1	01	NU	300	292

DSORG=PU RECFM=L LRECL=80
BLKSIZE=80 2ND ALLOCATION=20
EXT--FIRST--LAST-LENGTH
01 00740 C1015 00230
02 03640 C3655 00020

DODS-PROCLIB 69266 72350 PART 02 DODS01 C1 NU 120 35

DSORG=PU RECFM=FB LRECL=80
BLKSIZE=80 2ND ALLOCATION=20
EXT--FIRST--LAST-LENGTH
01 02323 C2415 00057
02 03660 C3682 00023

DODS-UTSLIN 70045 72350 PART 01 DODS01 01 NU 2 2

DSORG=PU RECFM=FB LRECL=80
BLKSIZE=160 2ND ALLOCATION=1

GSFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SLR=DODS01 UNIT=344

DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	FILE EXTENT	FILE SERIAL	VOL. SER.	SECURITY	TOTAL ALLOC	TRACKS USED
---------------	-----------------	---------------	--------------	----------------	----------------	--------------	----------	----------------	----------------

DODS-UTSLIN

EXT--FIRST--LAST-LENGTH
01 00438 C0439 00302

DODS-V4.CDT	69255	00000	SEQ.	01	DODS01	C1	NU	1	1
-------------	-------	-------	------	----	--------	----	----	---	---

DSORG=PS RECFM=V LRECL=790
BLKSIZE=800 2ND ALLOCATION=1
EXT--FIRST--LAST-LENGTH

01 01065 C1066 00001

DODS-V4.GOMUD	70030	74004	PART	02	DODS01	01	NU	260	246
---------------	-------	-------	------	----	--------	----	----	-----	-----

DSORG=PU RECFM=L LRECL=0
BLKSIZE=7254 2ND ALLOCATION=20

EXT--FIRST--LAST-LENGTH

01 00566 C0735 00240

02 01540 C1555 00020

DODS-V4.MESS	69250	00000	SEQ.	01	DODS01	C1	NU	49	49
--------------	-------	-------	------	----	--------	----	----	----	----

DSORG=PS RECFM=F LRECL=0
BLKSIZE=73 2ND ALLOCATION=0

EXT--FIRST--LAST-LENGTH

01 01066 C1134 00049

DODS-V4.TTF	69250	00000	SEQ.	01	DODS01	C1	NU	3	3
-------------	-------	-------	------	----	--------	----	----	---	---

DSORG=PS RECFM=V LRECL=796
 BLKSIZE=800 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 C1 00020 00022 00003

DODS.V5.CUT

70057 00000 SEQ. 01 DODS01 C1 NO 1 1

DSORG=PS RECFM=VS LRECL=796
 BLKSIZE=800 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 C1 03E2E C3E2E 00001

DODS.V5.MESS

70105 00000 SEQ. 01 DODS01 01 NO 50 49

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=73 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 C1 02903 02952 00000

DODS.V5.TTF

59255 00000 SEQ. 02 DODS01 C1 NO 3 3

DSORG=PS RECFM=VS LRECL=796
 BLKSIZE=800 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 C1 02975 02975 00001
 C2 03955 03956 00002

DODS.V6.CUT

70170 00000 SEQ. 01 DODS01 C1 NO 1 1

DSORG=PS RECFM=VS LRECL=796
 BLKSIZE=800 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 C1 01175 01175 00001

DODS.V6.MESS

70170 00000 SEQ. 01 DODS01 01 NO 50 49

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=73 2ND ALLOCATION=1
 EXT--FIRST---LAST-LENGTH
 01 02750 02805 00050

DODS.V6.TTF

70170 00000 SEQ. 02 DODS01 C1 NO 4 3

DSORG=PS RECFM=VS LRECL=796
 GSFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SER=DODS01 UNIT=544

DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	FILE EXTENTS	FILE SERIAL	VOL. SEQ.	SECURITY	TOTAL ALLOC	TRACKS USED
DODS.V6.TTF									

BLKSIZE=800 2ND ALLOCATION=2
 EXT--FIRST---LAST-LENGTH
 01 01656 01657 00002
 C2 03957 03958 00002

DODSET

59310 00000 PART 01 DODS01 01 NO 3 4

DSORG=PD RECFM=L LRECL=3672
BLKSIZE=7294 2ND ALLOCATION=2
EXT--FIRST--LAST-LENGTH
01 01020 01027 00003

D005PHOT

69256 79256 PART 01 D00501 01 NU 57 57

DSORG=PD RECFM=F LRECL=30
BLKSIZE=80 2ND ALLOCATION=5
EXT--FIRST--LAST-LENGTH
01 03883 03935 00057

FREQ

69256 00000 SEQ 01 D00501 01 NU 1 1

DSORG=PS RECFM=FB LRECL=80
BLKSIZE=3520 2ND ALLOCATION=1
EXT--FIRST--LAST-LENGTH
01 00018 00016 00001

LUNAR

70000 00000 SEQ 01 01SCH0 01 NU 64 64

DSORG=PS RECFM=F LRECL=1320
BLKSIZE=1320 2ND ALLOCATION=1
EXT--FIRST--LAST-LENGTH
01 02635 02602 00004

NEWDCLS

70154 00000 SEQ 02 D00501 01 NU 223 223

DSORG=PS RECFM=F LRECL=0
BLKSIZE=3520 2ND ALLOCATION=0
EXT--FIRST--LAST-LENGTH
01 02043 02127 00083
02 03240 03375 00140

NEWDCMAN

70256 00000 SEQ 01 D00501 01 NU 23 18

DSORG=PS RECFM=F LRECL=0
BLKSIZE=3520 2ND ALLOCATION=0
EXT--FIRST--LAST-LENGTH
01 00470 00454 00023

NEWDCOBS

70164 00000 SEQ 02 D00501 01 NU 253 252

DSORG=PS RECFM=F LRECL=0
BLKSIZE=3520 2ND ALLOCATION=0
EXT--FIRST--LAST-LENGTH
01 01668 01935 00252
02 02040 02042 00003

NEWPROT

70053 00000 PART 01 D00501 01 NB 5 5

DSORG=PD RECFM=F LRECL=30
BLKSIZE=80 2ND ALLOCATION=5
EXT--FIRST--LAST-LENGTH
01 03380 03384 00005

NULLPROT

69256 79256 PART 01 D00501 01 NU 1 1

DSORG=PU RECFM=F LRECL=80
 BLKSIZE=80 2ND ALLOCATION=1
 EXT--FIRST--LAST-LENGTH
 01 00019 00019 00001

GSFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SER=000501 UNIT=544

DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	EXTENTS	FILE SERIAL	VOL. SER.	SECURITY	TOTAL ALLOC	TRACKS USED
NUTCB DSORG=PS RECFM=F LRECL=400 BLKSIZE=400 2ND ALLOCATION=1 EXT--FIRST--LAST-LENGTH 01 01615 01632 00019 02 01633 01633 00001 03 02618 02618 00001 04 03556 03556 00001 05 03559 03559 00001 06 03827 03827 00001	70045	00000	SEQ.	06	03SCR1	01	NO	23	23
BLDE.GOMOD	70105	00000	PART	02	000501	01	NO	300	264

DSORG=PU RECFM=U LRECL=0 BLKSIZE=7294 2ND ALLOCATION=20 EXT--FIRST--LAST-LENGTH 01 03480 03735 00250 02 00140 00179 00040	70107	00000	SEQ.	01	000501	01	NO	1	1
PCONAF					****				
DSORG=PS RECFM=V LRECL=612 BLKSIZE=612 2ND ALLOCATION=0 EXT--FIRST--LAST-LENGTH 01 01137 01137 00001	70107	00000	SEQ.	01	000501	01	NO	1	1
PCONBE					****				
DSORG=PS RECFM=V LRECL=612 BLKSIZE=612 2ND ALLOCATION=0 EXT--FIRST--LAST-LENGTH 01 03837 03837 00001	59255	79255	PART	01	000501	01	NO	2	1
PROTODS					****				
DSORG=PU RECFM=F LRECL=80 BLKSIZE=80 2ND ALLOCATION=1 EXT--FIRST--LAST-LENGTH 01 01639 01639 00002	70045	00000	SEQ.	01	03SCR0	01	NO	3	6
SOLAR DSORG=PS RECFM=F LRECL=1320 BLKSIZE=1320 2ND ALLOCATION=1 EXT--FIRST--LAST-LENGTH 01 02470 02477 00003	70135	99350	NUT	01	000501	01	NO	10	
SYSCTLG					****				

DSORG= RECFM= LRECL=0
 BLKSIZE=0 2ND ALLOCATION=1
 EXT=FIRST--LAST-LENGTH
 01 02569 02578 00010

TESTPROT 70050 00000 PART 04 DDD501 01 NO 140 134

DSORG=PD RECFM=FB LRECL=30
 BLKSIZE=3200 2ND ALLOCATION=20
 EXT=FIRST--LAST-LENGTH
 01 03365 03464 00020
 02 02017 02030 00020
 03 02440 02459 00020
 04 03570 03589 00020

TSTAUSFL 70000 00000 SEQ. 01 DDD501 C1 NO 10 10

DSORG=PS RECFM=F LRECL=0
 OSFC SYSTEMS SUPPORT UTILITY - CASC ALLOCATION MAP
 CONTENTS-EN-VOLUME=SER=900501 UNIT=544

PAGE

DATA SET NAME
 BLKSIZE=120 2ND ALLOCATION=1
 EXT=FIRST--LAST-LENGTH
 01 03825 03836 00010

TSTCCAR 69350 00000 SEQ. 01 DDD501 01 NO 1 1

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=100 2ND ALLOCATION=0
 EXT=FIRST--LAST-LENGTH
 01 00017 00017 00001

TSLUC 69344 00000 SEQ. 01 DDD501 C1 NO 4 4

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=132 2ND ALLOCATION=0
 EXT=FIRST--LAST-LENGTH
 01 01171 01174 00004

V4DCARF 69250 00000 SEQ. 01 DDD501 C1 NO 4 4

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=100 2ND ALLOCATION=1
 EXT=FIRST--LAST-LENGTH
 01 00013 00016 00003

V4DLGC 69250 00000 SEQ. 01 DDD501 C1 NO 33 33

DSORG=PS RECFM=F LRECL=0
 BLKSIZE=132 2ND ALLOCATION=1
 EXT=FIRST--LAST-LENGTH
 01 01138 01170 00033

V7COT 70140 00000 SEQ. 01 DDD501 01 NO 1 1

DSORG=PS RECFM=VS LRECL=790
 BLKSIZE=800 2ND ALLOCATION=1
 EXT=FIRST--LAST-LENGTH
 01 00437 00437 00001

WORK .ARCL 10

DSORG=PS RECFM=L LRECL=7294
 BLKSIZE=7294 2ND ALLOCATION=5
 EXT=FIRST--LAST-LENGTH
 01 03817 03825 00010
 02 03950 03954 00005
 03 00455 00459 00005
 04 03550 03954 00005
 05 02806 02810 00005
 06 02811 02815 00005
 07 03465 03469 00005
 08 03470 03474 00005
 09 03475 03479 00005
 10 01600 01604 00005
 11 01605 01609 00005

WORK .DATA

09250 79250 PART 04 000501 01 NO 40 33

DSORG=PD RECFM=FB LRECL=30
 BLKSIZE=3200 2ND ALLOCATION=10
 EXT=FIRST--LAST-LENGTH
 01 02123 02192 00010
 02 02400 02469 00010
 03 03560 03969 00010
 04 03540 03545 00010

GSFC SYSTEMS SUPPORT UTILITY - BASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SEN=000501 UNIT=544

DATA SET NAME	DATE CREATED	DATE PURGE	FILE TYPE	EXTENTS	FILE SERIAL	VOL. SEQ.	SECURITY	TOTAL ALLOC	TRACKS USED
WORK .GOMOD	70210	00000	PART	01	000501	01	NO	300	264

DSORG=PD RECFM=L LRECL=0
 BLKSIZE=7294 2ND ALLOCATION=26
 EXT=FIRST--LAST-LENGTH
 01 01100 01479 00300

WORK .MODL 18

70173 00000 PART 01 000501 01 NO 100 16

DSORG=PD RECFM=L LRECL=0
 BLKSIZE=7294 2ND ALLOCATION=20
 EXT=FIRST--LAST-LENGTH
 01 01480 01579 00100

WORK .SNCL 18

70170 00000 PART 02 000501 01 NO 300 46

DSORG=PD RECFM=FB LRECL=80
 BLKSIZE=2000 2ND ALLOCATION=20
 EXT=FIRST--LAST-LENGTH
 01 02400 02659 00100
 02 02580 03059 00120

GSFC SYSTEMS SUPPORT UTILITY - BASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SEN=000501 UNIT=544

FIRST TRACK	LAST TRACK	LENGTH	EXTENT	DATA SET NAME
00001	00010	00010	01	VTCC
00011	00012	00002	01	DDDS.DML18

00013	00016	00004	01	VACCARF
00017	00017	00001	01	TSTCCAR
00018	00018	00001	01	FNEG
00019	00019	00001	01	NULLPRCT
00020	00022	00003	01	DDDS.V4.TTF

00023	00033	00011	01	DEFSCD
00034	00139	00106	01	AUTDES
00140	00175	00040	02	DDDS.GEMGB
00190	00375	00200	01	DDDS.GCMCU

00380	00436	00057	01	FREE SPACE * * *
00437	00437	00001	01	VZCLT
00438	00439	00002	01	DDDS.JTSLIN

00440	00469	00030	01	AACES
00470	00494	00025	01	NEWCCMAN
00495	00499	00005	03	*CRK.ARCLIB
00500	00739	00240	01	DDDS.V4.CCMCU

00740	01015	00280	01	DDDS.MCLL12

01020	01027	00008	01	DDDSSET

01028	01081	00057	02	FREE SPACE * * *
01085	01085	00001	01	DDDS.V4.CUT

01086	01134	00055	01	DDDS.V4.MESS

01135	01135	00001	01	DDDS.DUMMY

01136	01136	00001	02	DDDS.AILIEE

01137	01137	00001	01	FCCNAF
01138	01170	00033	01	VADLCG
01171	01174	00004	01	TS2LCG
01175	01175	00001	01	DDDS.V6.CUT

01175 01175 00003 01 00037410100

01175 01175 00001 03 FREE SPACE * * *
01180 01475 00000 01 WORK.SUMML
01180 01175 00100 01 WORK.MOCLIB
01580 01585 00020 01 ARC.WORKLIB
01500 01604 00005 10 WORK.ARCLIB
01605 01505 00005 11 WORK.ARCLIB
01610 01614 00005 04 FREE SPACE * * *
01515 01632 00010 01 NOTES
01633 01633 00001 02 NOTES
01534 01635 00002 01 FILES
01636 01565 00050 05 FREE SPACE * * *
01630 01667 00002 01 0003.V0.TTF

01668 01535 00002 01 NEW0005
01920 01529 00020 02 0003.V0.GLMML

01980 02035 00030 01 00005
02040 02042 00003 02 NEW0005
02043 02127 00055 01 NEW0005
02128 02102 00055 05 FREE SPACE * * *
02153 02152 00010 01 WORK.DAT
02153 02332 00130 02 AUTLES
02323 02411 00097 01 0003.V0.FRCLIB

02420 02435 00020 07 FREE SPACE * * *
GSFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

CONTENTS ON VOLUME=SER=000501 UNIT=000

FIRST TRACK	LAST TRACK	LENGTH	EXTENT	DATA SET NAME
02440	02455	00020	03	TESTPRCP
02460	02465	00010	02	WORK.DAT
02470	02477	00008	01	SOLAR
02478	02479	00002	05	FREE SPACE * * *
02430	02655	00150	01	WORK.SRCLIB
02550	02675	00017	01	ACDFL
02677	02755	00075	02	00005
02756	02805	00050	01	0003.V0.MESS ****

02806	02810	00005	05	WORK.ARCLIB
02811	02815	00005	06	WORK.ARCLIB
02816	02817	00002	05	FREE SPACE * * *
02818	02819	00001	03	NOTES
02819	02835	00020	02	TESTPRCP
02839	02802	00004	01	LUNAR
02803	02552	00050	01	0003.V0.MESS

02953	02966	00010	01	CO4MNV
02969	02978	00010	01	SYSCTL6
02979	02979	00001	01	DLDS.VB.TTF

02980	03099	00120	02	WCRK.SNCLIE
03100	03158	00059	10	FREE SPACE * * *
03159	03159	00001	01	ATCTD1
03160	03239	00000	02	DLDS.GEMCO

03240	03379	00140	02	NEWCELS
03380	03384	00005	01	NEWFRCT
03385	03404	00000	01	TESTPRUT
03465	03465	00005	07	WCRK.ANCLIE
03470	03474	00005	09	WCRK.ANCLIE
03475	03475	00005	09	WCRK.ANCLIE
03480	03739	00250	01	DLDS.GUMCO
03740	03754	00015	03	AUTLES
03755	03816	00002	03	CONLEB
03817	03826	00010	01	WCRK.ANCLIE
03827	03827	00001	06	NOTES
03828	03828	00001	01	DLDS.VB.CLT

03829	03838	00010	01	TOTAC5PL
03839	03839	00001	01	PCENDE
03840	03859	00020	02	DLDS.MULLIE

03850	03862	00023	02	DLDS.PFCLIE
-------	-------	-------	----	-------------

03883	03939	00057	01	DLDS.PRUT
-------	-------	-------	----	-----------

03940	03949	00010	04	WCRK.DATA
03950	03954	00005	04	WCRK.ANCLIE
03955	03957	00003	11	FREE SPACE * * *
03958	03958	00001	04	NOTES
03959	03959	00001	03	NOTES
03960	03969	00010	03	WCRK.DATA
03970	03989	00020	04	TESTPRUT
03990	03994	00005	02	WCRK.ANCLIE
03995	03996	00002	02	DLDS.VB.TTF

03997	03998	00002	02	DLDS.VB.TTF
-------	-------	-------	----	-------------

03999	03999	00001	12	FREE SPACE * * *
-------	-------	-------	----	------------------

USFC SYSTEMS SUPPORT UTILITY - CASE ALLOCATION MAP

PAGE

END OF UTILITY - CASES ARE MAPPED

```

IEF2851  SYS70274.T114519.RV000.GFRLECSS.LCDMCD  PASSED
IEF2851  VOL SER NUS= G1SCR9.
IEF2851  SYS70274.T114519.SV000.GFRLECSS.R0000169  SYSCUT
IEF2851  VOL SER NUS= G1SCR7.
IEF2851  SYS70274.T114519.SV000.GFRLECSS.R0000169  DELETED
IEF2851  VOL SER NUS= G1SCR7.
IEF2851  SYS70274.T114519.RV000.GFRLECSS.LIST  DELETED
IEF2851  VOL SER NUS= G1SCR5.
IEF2851  SYS70274.T114519.SV000.GFRLECSS.R0000170  DELETED
IEF2851  VOL SER NUS= G1SCR7.
-----JOB NBR= 690 STEP NBR= 04 GFRLECSS GC PGM=PGM=*.LC CARDS=00000 INITIATION TIME=21.10.55.94 DATE=10-01-70
-----CPU=000.1 I/O=000.1 CORE=000.1 CHARGE=000.00 STEP=04 GC TERMINATION TIME=21.11.13.26 DATE=10-01-70
----- I/O TIME BY DEVICE. DISK=*****.34,DRUM=*****.00,TAPE=*****.00,CELL=*****.00,UTHR=*****.00
-----STEP REGION SIZE=0115K MAXIMUM REGION SIZE USED=0000K PERCENT OF REGION USED=51
IEF2851  SYS70274.T114519.RV000.GFRLECSS.LCDMCD  DELETED
IEF2851  VOL SER NUS= G1SCR9.

```

```

----- CPU=000.1 I/O=000.1 CORE=000.1 CHARGE=000.20 JOB NBR=690 GFRLECSS SYSTEM=MVT-R16 (12-05-69) G1
----- I/O TIME BY DEVICE. DISK=*****.52,DRUM=*****.71,TAPE=*****.09,CELL=*****.00,UTHR=*****.00

```